

#3

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of:

HONG-SOON NAM, ET AL.

Serial No.

Filed:

For: **APPARATUS AND METHOD FOR PACKET  
SCHEDULING USING CREDIT BASED ROUND  
ROBIN - UTILITY**

J1000 U.S. PTO  
10/09/1870  
03/05/02

Honorable Commissioner of  
Patents and Trademarks  
Washington, D.C. 20231

**Request for Priority**

Sir:

Applicant respectfully requests a convention priority for the above-captioned application, namely  
Republic of Korea application number 01-75929 filed December 3, 2001.



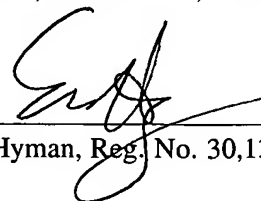
A certified copy of the document is being submitted herewith.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

Dated:

3/5/02



Eric S. Hyman, Reg. No. 30,139

12400 Wilshire Blvd., 7th Floor  
Los Angeles, California 90025  
Telephone: (310) 207-3800



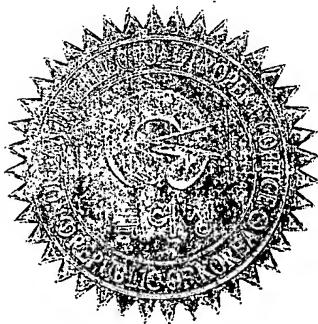
별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Intellectual  
Property Office.

출원번호 : 특허출원 2001년 제 75929 호  
Application Number PATENT-2001-0075929

출원년월일 : 2001년 12월 03일  
Date of Application DEC 03, 2001

출원인 : 한국전기통신공사 외 1명  
Applicant(s) KOREA TELECOM, et al.



2002 년 01 월 02 일

특 허 청

COMMISSIONER



CERTIFIED COPY OF  
PRIORITY DOCUMENT

**【서지사항】**

<b>【서류명】</b>	특허출원서
<b>【권리구분】</b>	특허
<b>【수신처】</b>	특허청장
<b>【참조번호】</b>	0002
<b>【제출일자】</b>	2001. 12. 03
<b>【국제특허분류】</b>	H04L 12/56
<b>【발명의 명칭】</b>	크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링 장치 및 방법
<b>【발명의 영문명칭】</b>	AN APPARATUS AND METHOD FOR SCHEDULING PACKETS BY USING A ROUND ROBIN BASED ON CREDIT
<b>【출원인】</b>	
<b>【명칭】</b>	한국전자통신연구원
<b>【출원인코드】</b>	3-1998-007763-8
<b>【출원인】</b>	
<b>【명칭】</b>	한국전기통신공사
<b>【출원인코드】</b>	2-1998-005456-3
<b>【대리인】</b>	
<b>【성명】</b>	손원
<b>【대리인코드】</b>	9-1998-000281-5
<b>【포괄위임등록번호】</b>	2001-038295-9
<b>【포괄위임등록번호】</b>	2001-044549-5
<b>【대리인】</b>	
<b>【성명】</b>	함상준
<b>【대리인코드】</b>	9-1998-000619-8
<b>【포괄위임등록번호】</b>	2001-038297-3
<b>【포괄위임등록번호】</b>	2001-044551-5
<b>【발명자】</b>	
<b>【성명의 국문표기】</b>	남홍순
<b>【성명의 영문표기】</b>	NAM, Hong Soon
<b>【주민등록번호】</b>	590513-1815113
<b>【우편번호】</b>	305-330

**【주소】** 대전광역시 유성구 지족동 열매마을 새미래아파트 816-102  
**【국적】** KR  
**【발명자】**  
**【성명의 국문표기】** 한만수  
**【성명의 영문표기】** HAN, Man Soo  
**【주민등록번호】** 690904-1666011  
**【우편번호】** 302-122  
**【주소】** 대전광역시 서구 둔산2동 주은오피스텔 1715  
**【국적】** KR  
**【발명자】**  
**【성명의 국문표기】** 전용일  
**【성명의 영문표기】** JUN, Yong Il  
**【주민등록번호】** 580905-1467311  
**【우편번호】** 305-707  
**【주소】** 대전광역시 유성구 신성동 한울아파트 106-1601  
**【국적】** KR  
**【발명자】**  
**【성명의 국문표기】** 이우섭  
**【성명의 영문표기】** LEE, Woo Seop  
**【주민등록번호】** 600611-1042411  
**【우편번호】** 305-755  
**【주소】** 대전광역시 유성구 어은동 한빛아파트 101-1702  
**【국적】** KR  
**【심사청구】** 청구  
**【취지】** 특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인  
 손원 (인) 대리인  
 함상준 (인)

**【수수료】**

【기본출원료】 20 면 29,000 원

【가산출원료】 19 면 19,000 원

【우선권주장료】 0 건 0 원

【심사청구료】 18 항 685,000 원

【합계】 733,000 원

【첨부서류】 1. 요약서·명세서(도면)\_1통

**【요약서】****【요약】**

본 발명의 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치 및 방법은 다수의 연결이 하나의 링크를 공유하는 고속 통신망에서 일시적으로 발생하는 혼잡에 대한 효율적인 서비스를 위하여 패킷 전송의 연결 속도에 비례하는 가중치를 미리 설정하고 이를 가용 크레딧으로 설정하여 패킷이 도착하면 상기 가용 크레딧 범위 내에서 필요한 크기의 크레딧을 갖는 토큰을 토큰 큐에 저장하고 가장 먼저 저장된 토큰이 지정하는 연결의 패킷을 서비스하도록 패킷을 스케줄링하는 장치 및 방법이다. 본 발명은, 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치에 있어서, 입력 패킷을 저장하는 패킷 풀과, 상기 패킷 풀에 저장된 입력 패킷의 연결 식별자, 상기 연결의 라운드 수 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 저장하는 토큰 큐, 및 상기 입력 패킷을 상기 패킷 풀로 전송하고 상기 패킷 풀에 저장된 패킷을 읽어들이어 상기 패킷에 대한 연결 식별자(ID) 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰 큐에 전송하며 상기 토큰 큐에 저장된 토큰이 지정하는 상기 패킷 풀의 패킷을 서비스하는 연결 관리부를 포함한다.

**【대표도】**

도 4

**【색인어】**

크레딧, 라운드 로빈, 토큰, 토큰 큐, 패킷, 연결, 패킷 풀, 가중치, 연결 관리부, 서비스, 연결식별자(ID)

**【명세서】****【발명의 명칭】**

크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치 및 방법{AN APPARATUS AND METHOD FOR SCHEDULING PACKETS BY USING A ROUND ROBIN BASED ON CREDIT}

**【도면의 간단한 설명】**

도 1은 본 발명에 따른 패킷 스케줄링이 적용되는 일반적인 ATM 교환기 혹은 라우터 구성의 일예이다.

도 2는 라운드 로빈의 갱신 과정에 따른 패킷 큐의 일실시예의 비교도로서,

도 2(a)는 종래의 라운드 로빈 갱신 과정에 따른 패킷 큐이며,

도 2(b)는 본 발명의 라운드 로빈 갱신 과정에 따른 패킷 큐이다.

도 3은 본 발명의 일실시예에 따른 패킷 스케줄링장치의 구성도이다.

도 4는 본 발명에 따른 패킷 도착 과정을 보이는 플로우차트이다.

도 5는 본 발명에 따른 패킷 출력 과정을 보이는 플로우차트이다.

도 6은 패킷 서비스에 대한 일실시예의 비교도로서,

도 6(a)는 종래의 가중 라운드 로빈에서 패킷 서비스의 일실시예이며,

도 6(b)는 본 발명에 따른 스케줄링장치의 패킷 서비스의 일실시예이다.

도 7은 본 발명에 따른 스케줄링장치에서 가중 크레딧에 따른 속도 제어의 일실시예이다.

\* 도면의 주요 부분에 대한 부호의 설명 \*

1 : 버퍼	2 : 스케줄러
3 : ATM교환기/라우터	20 : 패킷 큐
21 : 패킷	33 : 패킷 풀
34 : 연결 관리부	35 : 토큰 큐
36 : 토큰	37 : 연결 관리 테이블

### 【발명의 상세한 설명】

### 【발명의 목적】

### 【발명이 속하는 기술분야 및 그 분야의 종래기술】

<18> 본 발명은 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치 및 방법에 관한 것으로서 보다 상세하게는, 패킷 전송의 연결 속도에 비례하는 가중치를 미리 설정하고 이를 가용 크레딧으로 설정하여 패킷이 도착하면 상기 가용 크레딧 범위 내에서 필요한 크기의 크레딧을 갖는 토큰을 토큰 큐에 저장하고 가장 먼저 저장된 토큰이 지정하는 연결의 패킷을 서비스하도록 패킷을 스케줄링하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치, 방법 및 이를 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체에 관한 것이다.

<19> 일반적으로 통신망에서는 다수의 연결이 제한된 자원을 공유하므로 일시적인 혼잡이 발생할 수 있다. 혼잡 발생시 다수의 연결간 공평성(fairness)과 작은 레이턴시(low latency)를 제공하기 위하여 다양한 방법으로 스케줄링한다.

<20> 종래의 스케줄링 방법으로는 미국특허 제6,101,193호에 라운드 로빈방법이 개시되어 있다. 그러나, 이 방법은 시간 복잡도는 작지만 짧은 시간 공평성이 낮고



레이턴시가 큰 편이었다. 또한, 미국특허 제6,134,217호에 페어 큐잉 방법이 개시되어 있지만 공평성과 레이턴시는 양호하나 타임 스탬프에 따른 소팅으로 시간 복잡도가 연결 수에 따라 증가되는 문제점이 있었다.

<21> 고속 통신망에 사용되는 스케줄러는 연결간 공평성과 레이턴시를 제한하여야 할 뿐만 아니라 고속으로 운용되어야 하므로 시간 복잡도가 작아야 한다. 예를 들면, 10 Gbps 인터페이스에서 길이가 100 바이트의 패킷은 0.08 usec내에 처리되어야 한다.

<22> 부족 라운드 로빈(Deficit Round Robin; 참조논문- M.Shreedhar and George Varghese, Efficient Fair Queueing using Deficit Round Robin, SIGCOMM '95, pp. 231-241)과 가중 라운드 로빈(WRR;Weight Round Robin)은  $O(1)$ 의 시간 복잡도로 구현 가능하지만 레이턴시가 저하된다.

<23> 가중 라운드 로빈(WRR;Weight Round Robin)은 한 프레임에서 할당된 가중 크레딧 만큼을 서비스한 후 자신의 차례에서 자신에게 할당된 가중 크레딧 만큼 서비스 받을 수 있다. 자신이 서비스 받은 직후에 도착한 패킷은 다른 연결이 할당된 가중 크레딧 만큼 서비스 받은 후 서비스 받는다. 라운드 로빈에서 자신의 연결이 서비스받고 다음 서비스 받을 시간 간격을 라운드 크기라고 하면 공평성과 레이턴시는 라운드 크기에 의존한다. 라운드 크기는 백로그된 연결의 선두 패킷부터 패킷 크기의 합이 가중치 이하인 모든 백로그된 연결별 크기의 합이다. 따라서 라운드 크기는 가중치와 밀접한 관계를 갖는다. 가중 라운드 로빈에서 가중치  $W$ 는 최대 패킷 크기와 같거나 크게 설정되므로 연결이 많은 경우 라운드 크기는 큰 값이 될 수 있다. 인터넷에서 패킷의 크기는 수십 바이트에서 수 K바이트까지

크기가 다양하다. 따라서 가중치  $W$ 를 최대 패킷 크기보다 크게 설정할 경우 작은 패킷들은 버스트하게 서비스된다.

<24> 상기 부족 라운드 로빈은 한 라운드에 주어지는 양을 퀀텀이라고 하며 퀀텀을 최대 패킷 크기보다 작게 설정할 수 있다. 서비스가 되는 패킷의 지정하는 라운드 포인터가 자신의 차례가 되면 퀀텀 크기 만큼을 서비스 받는다. 이때 퀀텀보다 작은 패킷은 몇 개의 패킷이 서비스 받으며, 퀀텀보다 큰 패킷은 카운터 값이 패킷 크기와 같거나 클 때까지 다음 라운드의 퀀텀을 합하여 서비스 받는다. 퀀텀의 크기는 연결별로 한 라운드에 서비스 받을 양을 나타내므로 연결별로 다른 속도를 제공하는데 활용될 수 있다. 즉, 속도가 큰 연결은 퀀텀을 크게 설정하며 작은 연결은 작게 설정하여 서비스하면 퀀텀에 비례하는 속도로 서비스된다. 이 방법은 퀀텀의 크기보다 작은 패킷들이 도착하는 경우 자신의 차례가 되면 퀀텀에 해당하는 크기를 서비스하므로 연속하여 몇 개의 패킷들이 서비스 되어 버스트니스가 증가하게 된다. ATM 셀과 같이 모든 패킷의 크기가 동일한 연결의 속도를 제어하기 위하여 속도에 비례하게 퀀텀을 할당하여야 하므로 퀀텀 크기에 해당하는 만큼의 ATM 셀이 연속적으로 서비스 받아서 버스트(burst)하게 서비스되는 경우가 증가하게 된다.

<25> 이러한 문제를 극복하기 위한 한 예로서, 상기한 미국특허 제6,101,193호에 라운드 로빈 방법이 제안되었고, 참고문헌(Guo Chuanxiong, 'SRR: An  $O(1)$  Time Complexity packet scheduler for flows in multi-service packet networks', Proc. SIGCOMM '01, pp.211-222, Aug.2001)에 완화된 라운드 로빈 방법이 제안되었다.

<26> 상기 선출원된 특허출원의 라운드 로빈 방법에서는 2개의 선입선출(FIFO)로 운용되는 스케줄링 큐를 두고 각 연결의 선두(HOL; Head Of Line)에 대한 스케줄링 정보를 스케줄링 큐에 하나씩 저장한다. 이어, 패킷이 도착하면 그 연결의 HOL 패킷인지 확인하고 HOL 패킷이면 스케줄링 정보를 스케줄링 큐에 저장한다. 스케줄링 큐에는 2개의 큐가 있으며 각 큐는 가중치가 할당되어 있고, 가중치를 고려한 카운터 값보다 작은 패킷에 대한 스케줄링 정보는 현재 서비스되고 있는 큐에 저장하고, 패킷 카운터 값보다 크면 다른 큐에 저장한다. 서비스되고 있는 스케줄링 큐가 완전히 빌 때까지 계속 서비스하며, 그 큐에 백로그된 스케줄링 정보가 없을 때 다른 스케줄링 큐를 서비스한다. 이때 새로운 라운드가 시작된다.

즉, 현재 서비스되고 있는 스케줄링 큐가 전부 서비스되면 다른 스케줄링 큐로 스위치되어 새로운 라운드가 시작되고, 그 큐로부터 서비스되는 연결의 카운터 값을 가중치 만큼 증가시킨다. 스케줄링 큐는 FIFO로 서비스 되며, 패킷 크기와 카운터 값을 비교하여 패킷 크기가 카운터 값보다 작으면 서비스하고 서비스 받은 크기만큼 카운터 값을 감소시킨다. 반면 패킷 크기가 카운터 값보다 크면 카운터 값을 가중치 만큼 증가시키고, 스케줄링 정보를 서비스하지 않는 스케줄링 큐에 저장한다.

<27> 그러나, 상기한 방법은 라운드 로빈과 같이 버스트(burst)하게 서비스되는 것을 개선할 수 있으나, 연결별로 패킷 크기가 다른 경우에는 개선 효과가 감소되며, 패킷 크기가 가중치보다 큰 경우 현재 서비스되고 있는 스케줄링 큐에서 다음 스케줄링 큐로 반복하여 서비스해야 되는 문제점이 있었다. 상기 완화된 라운드 로빈은 고정 길이의 패킷에 대하여 미리 서비스 받을 연결의 순서를 정해놓고 미리

정해진 순서(sequence)에 따라 백로그된 셀이 있는 연결에 대하여 한 셀씩 서비스한다. 이로써, 연결 수(혹은 링크수)가 작은 경우 공평성과 레이턴시는 양호하지만, 연결 수가 많아지면 서비스될 순서인 시퀀스가 커지므로 큐에서 패킷을 출력하는 과정의 시간 복잡도가 증가되며 또한, 패킷 크기가 다양하면 공평성과 레이턴시가 저하되는 문제점이 있었다.

<28> 또한, 타임 스탬프(time stamp)기반의 스케줄링 방법으로는 참고문헌(S. J. Golestani, 'A self-clocked fair queueing scheme for high speed applications'. Proc. INFOCOM '94, pp.636-646, Apr.1994)에서의 셀프 클럭 페어 큐잉과, 가상 클럭(virtual clock) 및 참고문헌(D.Stidialis, A.Varma, 'Efficient Fair Queueing Algorithms for Packet-Switched Networks', IEEE/ACM Transactions on Networking, Vol.6, No.2, pp.175-185, April,1998)에서의 스타닝 포텐셜 페어 큐잉 등이 개시되어 있다. 그러나, 이러한 타임 스탬프 기반의 스케줄링 방법은 지연과 공평성은 양호하지만 타임 스탬프에 따른 순서 정렬을 위하여 적어도  $O(\log(N))$ 의 복잡도를 갖는다. 연결 수에 따라 시간 복잡도가 증가하므로 연결 수가 많고 속도가 빠른 고속 통신망에 구현하기 어려운 문제가 있었다.

#### 【발명이 이루고자 하는 기술적 과제】

<29> 본 발명은 상기한 문제를 해결하기 위한 것으로, 다수의 연결이 존재하고 각 연결의 서비스 속도가 다른, 다양한 크기의 패킷이 존재하는 망에서 각 연결 별로 가용 크레딧을 사용하여 속도를 제어하고 상기 가용 크레딧의 상태에 따라 도착하는 패킷을 서비스함으로써 공평성과 레이턴시를 개선되도록 패킷을 스케줄

링하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치, 방법 및 이를 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공하는데 그 목적이 있다.

#### 【발명의 구성 및 작용】

- <30>       상기한 목적을 달성하기 위한 본 발명에 따른 장치는, 각각의 서비스 속도를 갖는 복수개의 연결로 패킷을 송수신하는 고속 통신 망에서의 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치에 있어서,
- <31>       입력 패킷을 저장하는 패킷 풀;
- <32>       상기 패킷 풀에 저장된 입력 패킷의 연결 식별자(ID), 상기 연결의 라운드 수 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 저장하는 토큰 큐; 및
- <33>       상기 입력 패킷을 상기 패킷 풀로 전송하고 상기 패킷 풀에 저장된 패킷을 읽어들이어 상기 패킷에 대한 연결 식별자(ID), 상기 연결의 라운드 수(RN) 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰 큐에 전송하며 상기 토큰 큐에 저장된 토큰이 지정하는 상기 패킷 풀의 패킷을 서비스하는 연결 관리부를 포함한다.
- <34>       또한, 상기 목적을 달성하기 위한 본 발명에 따른 방법은, 각각의 서비스 속도를 갖는 복수개의 연결로부터 네트워크 스위치에 도착하는 복수개의 패킷을 수신하고 통신 링크로 패킷을 전송하는 고속 통신 망에서의 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법에 있어서,

- <35>      각 연결별로 서비스 속도에 비례하는 가중치(W)를 설정하고 상기 가중치를  
가용 크레딧(AC)으로 설정하는 제1 단계;
- <36>      입력되는 적어도 하나의 패킷을 수신하여 저장하는 제2 단계;
- <37>      상기 저장된 각 연결의 선두(HOL) 패킷의 나머지 크기(RSP)가 0이면 상기  
수신된 패킷의 크기(SP)를 상기 가용 크레딧(AC)의 크기와 비교한 결과에 따라  
상기 연결의 패킷에 대한 연결 식별자(ID), 상기 연결의 라운드 수(RN) 및 서비  
스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰 큐에 저장하는 제3 단  
계; 및
- <38>      상기 토큰 큐에 저장된 토큰이 지정하는 상기 저장된 패킷을 서비스하는  
제4 단계를 포함한다.
- <39>      또한, 상기 목적을 달성하기 위한 본 발명은, 대용량 프로세서를 구비하며  
각각의 서비스 속도를 갖는 복수개의 연결로부터 네트워크 스위치에 도착하는 복  
수개의 패킷을 수신하고 통신 링크로 패킷을 전송하는 고속 통신 망 시스템에,
- <40>      각 연결별로 서비스 속도에 비례하는 가중치(W)를 설정하고 상기 가중치를  
가용 크레딧(AC)으로 설정하는 제1 기능;
- <41>      입력되는 적어도 하나의 패킷을 수신하여 저장하는 제2 기능;
- <42>      상기 저장된 각 연결의 패킷의 크기(SP)를 상기 가용 크레딧(AC)의 크기를  
비교한 결과에 따라 상기 연결의 패킷에 대한 연결 식별자(ID), 상기 연결의 라  
운드 수(RN) 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰  
큐에 저장하는 제3 기능; 및

<43>      상기 토큰 큐에 저장된 토큰이 지정하는 상기 저장된 패킷을 서비스하는 제 4 기능을 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공한다.

<44>      본 발명은 패킷 도착시 가용 크레딧 및 도착 순서에 따라 순차적으로 토큰 큐에 저장하고 상기 저장된 순서에 따라 출력한다. 패킷이 도착하면 패킷 풀에 저장하고 가용 크레딧이 있으면 필요한 크기의 크레딧을 토큰에 실어 토큰 큐에 저장하며 가용 크레딧이 없으면 종료된다. 패킷의 출력은 패킷 큐의 선두(이하, HOL이라 함) 토큰에 대하여 토큰에 저장된 연결 식별자(ID)와 크레딧 크기를 참조하여 해당 연결의 HOL 패킷을 서비스한다. 패킷 풀에 저장된 연결 별 큐의 HOL 패킷이 HOL 토큰의 크레딧과 같거나 작으면 서비스하고, 그렇지 않으면 크레딧을 확보된 크레딧에 추가하고 가용 크레딧 만큼의 크레딧을 다시 토큰 큐에 저장한다. HOL 패킷이 서비스되고 패킷 풀에 해당 연결의 백로그된 패킷이 있으면 다시 할당된 필요한 만큼의 가용 크레딧을 토큰 큐에 저장한다. 이로써, 연결별로 가용 크레딧을 사용하여 속도를 제어하고 가용 크레딧의 상태에 따라 도착하는 패킷을 서비스함으로써 공정성과 레이턴시를 개선된다.

<45>      이하, 도면을 참조하여 본 발명을 보다 상세하게 설명한다.

<46>      도 1은 본 발명에 따른 패킷 스케줄링이 적용되는 일반적인 ATM 교환기 혹은 라우터 구성도의 일예이다. 도 1에 도시된 바와 같이, ATM 교환기 혹은 라우터(3)는  $n$  개의 입력 링크와  $n$ 개의 출력 링크로 구성되며 복수개의 버퍼(1)와 스케줄러(2)를 구비한다. 상기 각 입력 링크로 입력된 패킷 혹은 셀은 ATM 교환기 혹은 라우터(3)를 통해 출력 링크로 전송되며, 이 경우 동시에 동일 출력 링크로

출력되는 경우가 있으므로 버퍼(1)와 스케줄러(2)가 필요하다. 즉, 상기 버퍼(1)는 패킷 혹은 셀이 한 링크로 집중되는 경우에 대비하여 입력되는 패킷을 일시적으로 저장하기 위한 것이며, 상기 스케줄러(2)는 상기 버퍼(1)에 대기중인 패킷을 소정의 순서나 일정에 따라 요구되는 서비스 품질(QoS: Quality of Service)을 고려하여 서비스하기 위한 것이다. 통상적으로 통신망에서는 하나의 링크를 많은 연결(connection or flow)이 공유하여 사용하며, 각 연결이 요구하는 속도와 전송되는 패킷 크기가 다양하다. 따라서 각 연결이 요구하는 속도를 제공하고 레이턴시를 작게하는 스케줄링 장치 및 방법이 요구된다.

<47> 도 2는 라운드 로빈의 갱신 과정에 따른 패킷 큐의 일실시예를 도시한 것으로서, 도 2(a)는 종래의 라운드 로빈 갱신 과정에 따른 패킷 큐를, 도 2(b)는 본 발명의 라운드 로빈 갱신 과정에 따른 패킷 큐를 나타낸다. 패킷 큐(20)는 입력된 패킷을 출력할 때까지 일시적으로 저장하며 연결별로 관리한다. 도 2에 도시한 일실시예를 참조하면, 연결1 및 2의 가중치는 각각 800이며, 연결별 패킷 큐(20)에 패킷(21)들이 백로그 된 상태를 나타낸다. 각 패킷은  $P_{ij}$ 로 나타내며 상기  $i$ 는 연결을 나타내고 상기  $j$ 는 연결  $i$ 의 몇 번째 패킷인지를 나타낸다. 예를 들어,  $P_1^1$ 은 연결 1의 첫번째 패킷을,  $P_2^1$ 은 연결 2의 첫번째 패킷을 나타낸다. 상기 패킷 내의 숫자는 패킷 크기를 나타내며 단위는 바람직하게는 바이트로 설정한다.

<48> 도 2(a)에 도시된 바와 같이, 종래의 라운드 갱신은 백로그된 모든 연결에 그 연결의 가중치에 해당되는 패킷을 서비스한 후 다음 라운드로 갱신되며, 도면과



같은 경우에는 가중치가 800바이트로서 매 라운드마다 800 바이트씩 서비스를 받을 수 있다. 즉,  $P_1^1$ ,  $P_1^2$ ,  $P_1^3$ ,  $P_1^4$  및  $P_2^1$ 은 라운드 1에 서비스되며 나머지 패킷은 라운드 2에 서비스된다. 여기서, 상기 라운드 로빈 포인터는 서비스되는 패킷을 설정한다. 예를 들어, 도 2(a)에서와 같이,  $P_1^1$ 과  $P_2^1$ 이 동시에 도착한 경우 상기 라운드 로빈 포인터에서 가까운  $P_1^1$ 이 먼저 서비스된다.

<49> 반면, 도 2(b)에 도시된 바와 같이, 본 발명의 라운드 갱신은 가중 크레딧의 범위 내에서 하나의 패킷이 서비스 받으면 그 크레딧 만큼 라운드 윈도우가 이동하게 된다.  $P_1^1$ 과  $P_2^1$ 이 서비스 받고 현재 라운드 윈도우는 한 패킷씩 이동한다. 이 경우, 연결 1, 2의 입력 속도와 출력 속도가 동일하다고 가정하면, 도착 과정은  $P_1^1=P_2^1 > P_1^2 > P_1^3=P_2^2 > P_1^4 > P_1^5 > P_1^6 > P_1^7=P_2^3 > P_1^8$ 이다. 여기서, 상기 등호부호(=)는 동시에 도착하는 경우를 나타낸다.

<50> 상기에서 알 수 있듯이, 도 2(a)의 종래의 가중 라운드 로빈 혹은 가중 라운드 로빈 방법에서는 각 라운드의 패킷 순서대로 즉,  $P_1^1 > P_1^2 > P_1^3 > P_1^4 > P_2^1 > P_1^5 > P_1^6 > P_1^7 > P_1^8 > P_2^2 > P_2^3$  순으로 서비스된다. 특히, 상술한 종래의 미국특허 제6,101,193호의 경우에는 연결과 라운드의 패킷 순서대로 즉,  $P_1^1 > P_2^1 > P_1^2 > P_1^3 > P_1^4 > P_2^2 > P_1^5 > P_1^6 > P_2^3 > P_1^7 > P_1^8$  순으로 서비스를 받는다. 그러나, 도 2(b)의 본 발명의 라운드 로빈 방식에서는  $P_1^1 > P_2^1 > P_1^2 > P_1^3 = P_2^2 > P_1^4 > P_1^5 > P_1^6 > P_1^7 = P_2^3 > P_1^8$  순으로 서비스를 받을 수 있게 된다.

<51> 도 3은 본 발명의 일실시예에 따른 라운드 로빈장치의 구성도를 도시한 것이다. 도 3에 도시된 바와 같이, 본 발명에 따른 장치는 패킷 풀(33;packet pool), 연결 관리부(34;connection manager) 및 토큰 큐(35;token queue)로 구성된다. 입력되는 패킷(31)은 상기 연결 관리부(34)의 제어에 따라 상기 패킷 풀(33)에 저장되고 해당 연결의 라운드 수(RN;Round Number:이하, RN이라 함), 연결 식별자(ID) 및 서비스 받을 크레딧 값(CV;Credit value:이하, CV라 함)을 토큰(36)에 실어 상기 토큰 큐(35)에 저장시킨다. 상기 토큰 큐(35)는 상기 연결 관리부(34)의 제어에 따라 상기 토큰(36)을 저장하며 상기 패킷 풀(33)에 저장된 패킷을 선입선출(FIFO)로 서비스하도록 순서를 설정한다.

<52> 상기 패킷 풀(33)은 패킷을 저장하는 곳으로서 바람직하게는 버퍼로 구성되며, 바람직하게는 연계 리스트 등의 방법으로 연결별 큐를 갖는다. 상기 토큰 큐(35)는 하나의 선입선출(FIFO)로 서비스된다.

<53> 또한, 상기 연결 관리부(34)는 연결별로 필요한 연결 관리 테이블(37)을 가지며, 입력 패킷(31)을 입력하는 과정과 출력 패킷(32)을 출력하는 과정, 그리고 상기 토큰(36)을 상기 토큰 큐(35)에 저장하고 출력하는 과정을 관리한다. 상기 연결 관리부(34)는 상기한 연결 관리 테이블(37)을 관리하며, 연결 관리 테이블(37)은 연결별로 스케줄링에 필요한 파라미터를 나타낸다. 여기서는, 상기 파라미터는 생략한다.

<54> 하나의 연결은 고유한 연결 식별자(ID) 또는 연결 번호를 가지며 상기 연결 관리부(34)는 각 연결에 따라 가중치(W;weight), 가용 크레딧(AC;available credit:이하, AC라 함), HOL 패킷 크기(SP;size of packet:이하, SP라 함), 확보

된 크레딧(CC;confirmed credit:이하, CC라 함), 백로그된 패킷 크기(BS;backlog size:이하, BS라 함), HOL 패킷의 나머지 크기(RSP;residual size of packet:이하, RSP라 함) 등을 관리한다.

<55> 도 3에 도시된 연결 관리 테이블(37)의 일실시예를 참조하면, 각 연결 ID별로 관리되는 정보들을 저장되어 있다. 상기 가중치 크레딧 WC는 연결별로 서비스할 속도에 비례하여 설정되며, 패킷 크기와 무관하게 설정된다. 상기 가용 크레딧 AC는 가중치(W)에서 사용 가능한 크기를 나타내며,  $AC \leq W$  이다. 예를 들어, 연결 i에서 가중치 크레딧 WC가 400 이면 최초 초기화된 상태에서는 사용가능한 크기 즉, 가용 크레딧의 크기는 400이 된다. 이후에 입력 패킷 크레딧의 크기가 200이면 가중치 크레딧 400-패킷의 크기 200하면 가용 크레딧 AC는 200이 된다. 따라서, 최초의 가용 크레딧에서 200만큼 줄어든 크기의 가용 크레딧이 설정된다.

<56> 또한, 패킷 크기 SP는 패킷 풀(33)에 대기 중인 연결 i의 HOL 패킷의 크기를 나타낸다. 상기 가용 크레딧 AC가 상기 패킷 크기 SP보다 작은 경우 한번의 가용 크레딧으로 그 패킷을 서비스할 수 없다. 이러한 경우 여러 번의 가용 크레딧 AC를 합하여 사용할 수 있는데 이때 가용 크레딧 AC를 확보된 크레딧 CC에 추가한다. 상기한 확보된 크레딧 CC는 토큰 큐(35)의 HOL 토큰(36)으로부터 받은 가용 크레딧 AC중 서비스 되지 않은 크레딧을 말하며, HOL 패킷 크기 SP와 비교하여 확보된 크레딧 CC가 크면 HOL 패킷을 서비스하고, 다시 다음 HOL 패킷이 가용 크레딧 보다 작은 경우에는 그 패킷들을 서비스 한다. 즉,  $CC \geq SP$ 이면 HOL 패킷을 서비스하고 반면,  $CC < SP$ 이면 상기 확보된 크레딧 CC를 저장하고 다음

토큰(36)의 가용 크레딧 AC를 기다린다. 연결  $i$ 의 백로그된 패킷 크기 BS는 패킷 풀(33)에 대기중인 연결  $i$ 의 전체 패킷 크기를 나타낸다. 이는 가용 크레딧 보다 큰 패킷이 입력되면 그 차이만큼이 백로그 크기 BS로 저장되는 것이다. 차후에 가용 크레딧 보다 작은 크기의 패킷이 입력되면 상기 BS에 저장된 백로그된 패킷이 서비스된다.

<57> 도 4는 본 발명에 따른 패킷 도착 과정을 보이는 플로우차트이다. 연결  $i$ 의  $j$ 번째의 패킷  $P_{ij}$ 가 스케줄러(2)에 도착하면(S401), 연결 관리부(34)는  $P_{ij}$ 를 패킷 풀(33)에 저장하고, 패킷 크기  $SP_{ij}$ 를  $P_{ij}$ 의 크기로, 백로그 크기  $BS_i$ 를  $BS_i + SP_{ij}$ 로 설정한다(S402). 이어, 상기 연결 관리부(34)는 연결 관리 테이블(37)에서 해당 연결  $i$ 의 가용 크레딧  $AC_i$ 가 0이거나 HOL패킷의 나머지 패킷 크기  $RSP_i$ 가 0이 아닌지를 판단한다(S403). 상기 단계(S403)의 판단결과 상기 연결  $i$ 의 가용 크레딧  $AC_i$ 가 0이거나 HOL패킷의 나머지 패킷 크기  $RSP_i$ 가 0이 아니면 이 과정은 종료하고 반대로, 상기 단계(S403)의 판단결과, 상기  $AC_i$ 가 0이 아니거나 HOL 패킷의 나머지 패

킷 크기  $RSP_i$ 가 0이면 이후의 단계(S404)로 진행하여 상기  $AC_i$ 가  $SP_i^j$ 보다 크거나 같은지를 판단한다(S404). 상기 단계(S404)의 판단결과, 상기  $AC_i$ 가 상기  $SP_i^j$ 보다 크거나 같으면 상기 연결  $i$ 의 서비스 받을 크레딧 값  $CV$ 를 상기 패킷 크기  $SP_i^j$ 로, 상기 가용 크레딧  $AC_i$ 를  $AC_i - SP_i^j$ 로, 상기  $RSP_i$  및 연속 라운드 수  $RN$ 을 0으로, 연결 식별자  $ID$ 를  $i$ 로 설정하고 상기 설정된  $RN, CV, ID$ 에 대응하는 토큰  $T<RN, CV, ID>$ 를 토큰 큐(35)에 저장한 후(S405) 종료한다. 그러나, 상기 단계(S404)의 판단결과, 상기  $AC_i$ 가  $SP_i^j$ 보다 작으면 이후의 단계(S406)로 진행하여  $SP_i^j - AC_i$ 가  $W_i$ 보다 작거나 같은지를 판단한다(S406). 상기 단계(S406)의 판단결과, 상기  $SP_i^j - AC_i$ 가  $W_i$ 보다 작거나 같으면 연결  $i$ 의 서비스 받을 크레딧 값  $CV = AC_i$ 로, 상기 연결  $i$ 의 HOL 패킷의 나머지 크기  $RSP_i = SP_i^j - AC_i$ 로,  $RN = 1$ ,  $ID = i$ 로 설정하고 상기 설정된  $RN, CV, ID$ 에 대응하는 토큰  $T<RN, CV, ID>$ 를 토큰 큐(35)에 저장하며(S407), 이후에 상기  $AC_i = 0$ 으로 설정하고(S408) 종료한다.

<58> 그러나, 상기 단계(S406)의 판단결과, 상기  $SP_i^j - AC_i$ 가  $W_i$ 보다 크면 연결  $i$ 의 서비스 받을 크레딧 값  $CV$ 를  $CV = AC_i$ 로, 라운드 수  $RN = \lceil (SP_i - AC_i - 1) / W_i \rceil$ 로, 상기 연결  $i$ 의  $RSP_i = SP_i - AC_i - (RN - 1)W_i$ ,  $ID = i$ 로 설정하고, 토큰  $T<RN, CV, ID>$ 를 토큰 큐(35)에 저장한 후(S409),  $AC_i = 0$ 으로 설정하고(S410) 종료한다.

<59> 여기서, 상기 라운드 수  $RN$ 은, 패킷 크기  $SP$ 가 가중치( $W$ )보다 커서 여러 개의 가용 크레딧  $AC$ 가 필요한 경우 토큰(36)이 토큰 큐(35)를 여러 번 반복하게 되는데 이때 연결 관리부(34)가 연결 관리 테이블(37)을 참조하지 않고 HOL 토큰의 라운드 수만 확인하고 토큰 큐(36)에 저장할 것인가 빨리 판단하기 위함이다. 즉, 상기  $RN = 2$  이상이면 상기 연결 관리부(34)는 상기 연결 관리 테이블(37)의

값들을 참조할 필요없이 RN을 1만큼 감소시켜 토큰을 토큰 큐에 저장하며, 상기 RN이 1 이면 RN을 다시 0으로 설정하고 상기 연결 관리 테이블(37)의  $RSP_i$  값을 참조하여 CV값을 생산하며, 상기 RN이 0이면 상기 연결 관리 테이블(37)에 따라 패킷을 서비스한다.

<60> 도 5는 본 발명에 따른 패킷 출력 과정을 보이는 플로우차트이다. 상기한 토큰 큐(35)에 저장된 토큰(36)의 HOL 패킷이 출력된 후 다음 출력할 패킷을 검색한다(S501). 상기 토큰 큐(35)에 대기 중인 토큰(36)이 있는지 확인한다(S502). 상기 단계(S502)의 판단결과 대기 중인 토큰이 없으면 종료하고 반대로, 대기 중인 토큰이 있으면 상기 토큰 큐(35)의 HOL 토큰(36)을 서비스한다. 상기 HOL 토큰을 토큰  $T_{HOL} \langle RN, CV, ID \rangle$ 라고 하고 상기 HOL 토큰에 대한 라운드 수를  $T_{HOL}.RN$ 이라 하면, 상기  $T_{HOL}.RN$ 이 1 보다 큰지를 판단한다(S503). 상기 단계(S503)의 판단결과 상기  $T_{HOL}.RN$ 이 1 보다 크면 상기 토큰 큐(35)에 저장된 토큰(36)의 라운드 수 RN을  $RN-1$ 로 하나 감하여 다시 토큰 큐  $T \langle RN, CV, ID \rangle$ (35)의 마지막에 저장한 후(S504), 상기 단계(S501)로 되돌아가 상기 과정을 반복한다. 그러나, 상기 단계(S503)의 판단결과  $T_{HOL}.RN$ 이 1 보다 작거나 같으면 이후의 단계(S505)에서 상기  $T_{HOL}.RN$ 이 0인지 판단한다(S505). 상기 단계(S505)의 판단결과,  $T_{HOL}.RN$ 이 0이 아니면  $CV = \min(AC_i, BS)$ 로,  $CC_i = SP_i - RSP_i$ 로,  $RN = 0$ 으로 설정하고 상기 설정된  $RN, CV, ID$ 에 대응하는 토큰  $T \langle RN, CV, ID \rangle$ 을 토큰 큐(35)에 저장한 후(S506),  $RSP_i = 0$ 으로 재설정하고(S507) 종료한다. 그러나, 상기 단계(S505)의 판단결과, 상기  $T_{HOL}.RN$ 이 0이면 상기 토큰  $T_{HOL}$ 의 연결 식별자 ID를  $i$ 로, CV를

$T_{HOL}$ 의 CV로, 연결  $i$ 의 확보된 크레딧  $CC_i$ 를  $CC_i+CV$ 로, 연결  $i$ 의 가용 크레딧  $AC_i$ 를  $AC_i+CV$ 로 설정한다(S508).

<61> 계속하여, 연결  $i$ 의  $CC_i$ 가  $SP_i$ 와 같거나 큰지를 판단한다(S509). 상기 단계(S509)의 판단결과, 상기  $CC_i$ 가  $SP_i$ 와 같거나 크면 연결  $i$ 의 HOL 패킷을 서비스하고  $BS_i$ 를  $BS_i-SP_i$ 로,  $CC_i$ 를  $CC_i-SP_i$ 로 설정한다(S510). 이후에, 다시 패킷 풀(33)에 연결  $i$ 의 패킷이 있는지를 판단하여(S512), 대기중인 상기 연결  $i$ 의 패킷이 없으면  $CC_i$  및  $RSP_i$ 를 0으로,  $AC_i$ 를  $W_i$ 로 설정한 후(S515) 종료하고, 반대로 대기중인 상기 연결  $i$ 의 패킷이 있으면  $SP_i$ 를 연결  $i$ 의 HOL 패킷 크기로 설정한 후(S514), 상기 단계(S509)로 되돌아가 해당하는 과정을 반복한다.

<62> 그러나, 상기한 단계(S509)의 판단결과, 상기  $SP_i$ 가 상기  $CC_i$ 보다 크면 이후의 단계(S511)로 진행하여  $RSP_i$ 가 0인지를 판단한다(S511). 상기 단계(S511)의 판단결과 상기  $RSP_i$ 가 0이 아니면 종료하고 반대로 상기 단계(S511)의 판단결과 상기  $RSP_i$ 가 0이면 이후의 단계(S513)으로 진행하여  $AC_i$ 가  $SP_i-CC_i$ 보다 크거나 같은지를

판단한다(S513). 상기 단계(S513)의 판단결과, 상기  $AC_i$ 가  $SP_i-CC_i$ 보다 크거나 같으면 서비스 받을 크레딧 값  $CV=\min(W_i,BS_i)$ 로,  $RSP_i=0$ 으로,  $RN=0$ ,  $ID=i$ 로 설정하고 상기  $RN,CV,ID$ 에 대응하는 토큰  $T<RN,CV,ID>$ 을 토큰 큐에 저장한 후(S516), 연결  $i$ 의 가용 크레딧  $AC_i=AC_i-CV$ 로 재설정하고(S518) 종료된다. 반대로, 상기 단계(S513)에서  $AC_i$ 가  $SP_i-CC_i$ 보다 작은 것으로 판단되면 상기 연결  $i$ 의 서비스 받을 크레딧 값  $CV=AC_i$ 로,  $RN=\lceil (SP_i-CC_i-1)/W_i \rceil$ 로,  $RSP_i=SP_i-CC_i-AC_i-(RN-1)W_i$ 로,  $ID=i$ 로 설정하고 상기 설정된  $RN,CV,ID$ 에 대응하는 토큰  $T<RN,CV,ID>$ 를 토큰 큐(35)에 저장한 후(S517), 다시  $AC_i=0$ 으로 재설정하고(S519) 종료된다. 상기과 같은 과정은 HOL 패킷 크기보다 작고  $W_i$ 보다 작은 크레딧의 토큰이 발생하는 것을 예방하기 위한 것이다.

<63> 도 6은 패킷 서비스에 대한 일실시예의 비교도로서, 도 6(a)는 종래의 가중라운드 로빈(WRR)에서 패킷 서비스의 일실시예를 도시한 것이고 도 6(b)는 본 발명에 따른 스케줄러에서 패킷 서비스의 일실시예를 도시한 것이다.

<64> 도 6에서는 4개의 입력이 하나의 링크로 출력되는 경우로서 각각의 입력 스트림과 출력 스트림의 속도가 동일하다고 가정한다. 또한, 4개의 연결이 동시에 도착하고 있으며 패킷 크기에 따라 하나의 완전한 패킷이 도착하는 경우 즉, 마지막 바이트가 도착하는 경우에 패킷이 도착하였다고 가정한다. 도 6(a) 및 도 6(b)에 도시된 바와 같이, 각 연결 1,2,3,4의 가중 크레딧  $W$ 를 각각  $W_1 = 400$ ,  $W_2=800$ ,  $W_3=500$ ,  $W_4=500$ 으로 가정한다. 스케줄러에 도착하는 패킷은  $P_1^1=P_4^1 > P_3^1 > P_1^2 > P_4^2 > P_3^2 > P_1^3 > P_2^1$  순으로 도착된다.



<65> 도 6(a)와 같이, 종래의 가중 라운드 로빈(WRR) 방식인 경우에는  $P_1^1 > P_1^2 > P_3^1 > P_4^1 > P_4^2 > P_1^3 > P_2^1 > P_3^2$  순으로 서비스되지만, 도 6(b)과 같이 본 발명의 경우에는  $P_1^1 > P_4^1 > P_3^1 > P_1^2 > P_4^1 > P_3^2 > P_1^3 > P_2^1$  순으로 서비스된다. 단, 도 6(b)에서  $P_1^1$  과  $P_4^1$  은 동시에 도착한 경우로서 라운드 로빈 포인터 진행 방향으로 가까이 있는 연결부터 서비스 된다. 토큰 큐(35)에는 패킷이 도착할 때 도 4에서 설명한 바와 같이 가용 크레딧  $AC_i$ 와 패킷 크기  $SP_i$ 로부터 서비스 받을 크레딧 값  $CV$ 를 계산한 다음 토큰(36)을 토큰 큐(35)에 저장한다. 상기 저장된 토큰(36)은 HOL 토큰부터 서비스 받는다. 이와 같이 본 발명에 따르면 도 6(b)는 패킷 도착 순서와 동일하게 서비스 되는 경우를 나타낸다. 만약, 동시에 도착하는 경우에는 라운드 포인터의 진행 방향으로 가까이 위치한 연결의 패킷부터 서비스 받는다. 이와 같은 동일 조건에서 도 6(a)의 경우에는  $P_1^2$ 가  $P_4^1$ 과  $P_3^1$ 보다 늦게 도착하지만 먼저 서비스 받은 경우를 나타낸다. 따라서,  $P_4^1$ 은  $P_1^2$ 와  $P_3^1$ 이 전송되는 시간 만큼 지연되어 서비스 받게 되므로 레이턴시가 증가하게 된다.

<66> 도 7은 본 발명에 따른 스케줄러에서 가중 크레딧에 따른 속도 제어의 일실시예를 도시한 것이다. 도 7에 도시된 바와 같이, 연결 1, 2는 동일한 크기의 패킷이 도착하고 연결 3, 4는 패킷이 도착하지 않은 경우를 나타낸다. 연결 1의 가중 크레딧  $W_1$ 은 200이며, 연결 2의 가중 크레딧  $W_2$ 는 100인 경우이며, 이 경우에도 모든 링크의 속도는 동일하다고 가정하면 패킷 도착 순서는  $P_1^1 = P_2^1 > P_1^2 = P_2^2 > P_1^3 > P_1^4$  순이다. 그러나 상기 가중 크레딧  $W_1$ 과  $W_2$ 가 서로 다르므로 출력되는 패킷은  $P_1^1 > P_1^2 > P_2^1 > P_1^3 > P_1^4 > P_2^2$  순으로 출력된다. 연결 1의 가중

크레딧  $W_1$ 이 200이므로 CV가 최대 200이 되고 이로써,  $P_1^1$ 과 같이 길이가 200인 패킷이 도착하면  $T<0,200,1>$ 의 토큰(36)을 발생하여 토큰 큐(35)에 저장하며, 연결 2의 가중 크레딧  $W_2$ 는 100이므로 CV가 최대 100이 되고, 이로써  $P_2^1$ 과 같이 패킷 길이 200인 패킷이 도착하면 연결 2의 토큰  $T<1,100,2>$ 을 토큰 큐(35)의 마지막 저장하고, 다음 라운드에서 토큰  $T<0,100,2>$ 를 저장하여 서비스 한다. 여기서, 상기 연결 2는 2번의 토큰을 받아서 하나의 패킷을 서비스해야 되므로 연결 1에 비하여 0.5배의 속도로 서비스 받는다. 토큰(36)에 허용되는 최대 크레딧 값 CV는 가중 크레딧  $W_i$ 보다 작거나 같기 때문이다. 토큰 큐(36)에는 도 7에 도시된 바와 같은 토큰(36)이 발생하여 저장되고 서비스된다.

<67> 상술한 바와 같이, 패킷이 도착하면 가용 크레딧이 있는 경우에는 토큰 큐에 패킷 크기와 가용 크레딧 만큼의 크레딧을 토큰에 실어 토큰 큐에 전달한다. 임의의 시간  $t$ 에 백로그된 연결을  $B(t)$ 라고 하면 토큰 큐에 저장된 토큰의 크레딧의 합  $T\_Credit$ 은 하기 수식 2와 같이 된다.

<68> [수식 2]

$$<69> \quad T-Credit = \sum_{j \in B(t)} Min(W_i, BSi)$$

<70> 새로운 연결의 패킷  $P_{i,j}$ 가 도착하면 그 패킷의 크레딧 값  $CV=Min[W_i, SP_{i,j}]$ 를 토큰  $T_i$ 에 실어 저장한다. 상기 토큰  $T_i$ 는 상기  $T\_Credit$ 을 서비스 한 후에 서비스 받을 수 있으나 상기  $T\_Credit$ 에서 토큰을 서비스하더라도 그 해당

연결의 확보된 크레딧  $CC_i$ 가 상기 연결의 HOL 패킷의 크기  $SP_i$ 보다 작으면 패킷을 출력하지 않는다. 따라서 새로운 연결의 토큰이 토큰 큐에 저장된 후 다음 라운드에 서비스 받을 연결 집합을  $SC_n$  이라고 하고, 라운드 크기를  $F_n$ 이라고 하면 상기  $SC_n$ 과 상기  $F_n$ 은 하기 수식 3과 같이 된다.

<71> [수식 3]

$$<72> \quad SC_n \leq [i | (CC_i + W_i) \geq SP_i, CC_i < SP_i, \forall i \in B]$$

$$<73> \quad F_n \leq \sum_{j \in SC(n)} \sum_{k=1}^m SP_j^k, m = [\max l \mid \sum_{k=1}^l SP_j^k < (CC_j + W_j)]$$

<74> 상기 수식 3에서 상기  $SC_n$ 은  $n-1$ 번째 라운드에서는 서비스 되지 않고  $n$  번째 라운드에서 서비스되는 경우를 나타낸다. 즉,  $CC_i < SP_i = CC_i - W_i$ 를 나타낸다. 또한,  $m$ 은 연결  $j$ 에 대하여 현재 라운드에서 서비스할 수 있는 패킷 수를 나타내며  $W_i$ 가 0으로 수렴하면  $SC_n$  및  $F_n$ 도 0으로 수렴된다.  $W_i$ 가 0으로 접근하면 서비스 될 연결 집합  $SC$ 의 시간적 분포  $SC_1, SC_2, \dots, SC_n$  는 가중 페어 큐잉(WFQ; Weighted Fair Queueing)에서 가상 종료 시간 순서로 수렴하게 된다. 상기  $SC_n$  는 임의의 연결  $i$ 의 HOL 패킷의 가상 서비스 종료 시간이  $CC_i$ 와  $CC_i + W_i$  사이에 있는 연결의 집합보다 같거나 작으므로  $W_i$ 가 0으로 접근하게 되면 가상 페어 큐잉(WFQ)의 공평성과 레이턴시 특성으로 접근하게 된다. 라운드 크기  $F_n$ 은  $n$  번째 라운드에서 가상 종료 시간이  $CC_i$ 와  $CC_i + W_i$  사이에 있는 연결의 패킷 크기의 합과 같거나 작다.  $W$ 를 작게 설정하면 공평성과 레이턴시는 개선되지만  $RN$ 이 증

가하여 토큰이 큐에서 무한히 반복될 수 있으므로 패킷 크기를 고려하여 선정할 필요가 있다. 상기 방법은 종래의 라운드 로빈과 달리 가중치  $W$ 를 연결별 서비스 속도에 따라 비례하게 설정할 수 있으므로 다수의 연결에 대하여 다양한 속도와 다양한 크기의 패킷에 대하여 공평성과 레이턴시를 개선할 수 있다. 또 시간 복잡도가  $O(1)$ 이므로 타임 스탬프 방식에 비하여 연결 수에 따른 확장성이 용이하다.

<75>       이상에서 설명한 본 발명은 전술한 실시예 및 첨부된 도면에 의해 한정되는 것이 아니고 본 발명의 기술적 사상을 벗어나지 않는 범위 내에서 여러 가지 치환, 변환 및 변경이 가능하다는 것이 본 발명이 속하는 기술분야의 통산의 지식을 가진 자에게 있어 명백할 것이다.

#### 【발명의 효과】

<76>       이상에서 설명한 바와 같이, 본 발명은 시간 복잡도가  $O(1)$ 인 고속 스케줄러용으로서 다양한 크기의 패킷과 다수의 연결에 대하여 종래의 라운드 로빈 방식에 비해 공평성과 레이턴시를 개선할 수 있다. 또한, 시간 복잡도가  $O(1)$ 이므로 고속 통신망에 사용되는 비동기전송모드(ATM) 스위치, 라우터, 통신 단말기 등에 용이하게 활용될 수 있다.

**【특허청구범위】****【청구항 1】**

각각의 서비스 속도를 갖는 복수개의 연결로 패킷을 송수신하는 고속 통신망에서의 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치에 있어서,

입력 패킷을 저장하는 패킷 풀;

상기 패킷 풀에 저장된 입력 패킷의 연결 식별자(ID), 상기 연결의 라운드 수 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 저장하는 토큰 큐; 및

상기 입력 패킷을 상기 패킷 풀로 전송하고 상기 패킷 풀에 저장된 패킷을 읽어들이어 상기 패킷에 대한 연결 식별자(ID), 상기 연결의 라운드 수(RN) 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰 큐에 전송하며 상기 토큰 큐에 저장된 토큰이 지정하는 상기 패킷 풀의 패킷을 서비스하는 연결 관리부를 포함하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

**【청구항 2】**

제 1항에 있어서,

상기 토큰 큐는 선입선출(FIFO)로 서비스되는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

**【청구항 3】**

제 1항에 있어서,

상기 토큰에 설정된 서비스 받을 크레딧 값(CV)은 상기 토큰 큐의 선두 (HOL) 패킷 크레딧 값인 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

#### 【청구항 4】

제 1항에 있어서, 상기 연결 관리부는,

연결별로 가중치(W)를 설정하고 이를 가용 크레딧(AC)으로 설정하며 상기 설정된 가용 크레딧(AC)보다 작은 크기의 패킷이 도착하면 해당 연결 식별자 (ID), 상기 연결의 라운드 수(RN) 및 상기 패킷 크기의 크레딧 값(CV)으로 갖는 토큰을 발생시켜 상기 토큰 큐에 전송시키고 상기 설정된 가용 크레딧(AC)보다 큰 크기의 패킷이 도착하면 상기 패킷 크기와 상기 가용 크레딧(AC)의 차가 상기 가중치(W)보다 작은 경우에는 해당 연결 식별자, 상기 연결의 라운드 수(RN) 및 상기 가용 크레딧(AC)을 상기 크레딧 값(CV)으로 갖는 토큰을, 큰 경우에는 해당 연결 식별자, 상기 연결의 라운드 수(RN) 및 상기 가중치(W)를 상기 크레딧 값 (CV)으로 갖는 토큰을 발생시켜 상기 토큰 큐에 전송하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

#### 【청구항 5】

제 4항에 있어서,

상기 설정된 가용 크레딧(AC)보다 작은 크기의 패킷이 도착하면 상기 가용 크레딧(AC)을 {상기 가용 크레딧(AC)-패킷의 크기}로, 라운드 수(RN)를 0으로 재 설정하고, 상기 설정된 가용 크레딧(AC)보다 큰 크기의 패킷이 도착하면 상기 패

킷 크기와 상기 가용 크레딧(AC)의 차가 상기 가중치(W)보다 작은 경우에는 선두(HOL) 패킷의 나머지 크기(RSP)를 {상기 패킷의 크기-상기 가용 크레딧(AC)}로, 상기 라운드 수(RN)를 0으로 재설정 후 상기 가용 크레딧(AC)을 0으로 재설정하고, 반대로 큰 경우에는 상기 라운드 수(RN)를  $\lceil \{\text{패킷의 크기} - \text{확보된 크레딧(CC)} - 1\} / \text{가중치(W)} \rceil$ 으로, 상기 선두(HOL) 패킷의 나머지 크기(RSP)를 {상기 패킷의 크기-상기 가용 크레딧(AC)-(상기 라운드 수(RN)-1)  $\times$  가중치(W)}로 재설정 후 상기 가용 크레딧(AC)을 0으로 재설정하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

#### 【청구항 6】

제 1항에 있어서,

상기 가용 크레딧(AC)이 상기 가중치(W) 및 선두(HOL) 패킷의 크기(SP)보다 작으면 토큰을 발생하지 않고 다음 토큰의 발생을 기다려 작은 토큰을 통합하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

#### 【청구항 7】

제 1항에 있어서, 상기 연결 관리부는,

각 연결별로 연결 식별자(ID), 가중치(W), 가용 크레딧(AC), 선두(HOL) 패킷 크기(SP), 확보된 크레딧(CC), 백로그된 패킷 크기(BS), 선두(HOL)의 나머지 패킷 크기(RSP)의 파라미터를 관리하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

**【청구항 8】**

제 1항에 있어서, 상기 연결 관리부는,

동일 연결에 대하여 적어도 하나의 토큰을 가지며 상기 가용 크레딧(AC) 범위 내에서 도착하는 패킷의 순서에 딸 상기 패킷을 서비스하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링장치.

**【청구항 9】**

각각의 서비스 속도를 갖는 복수개의 연결로부터 네트워크 스위치에 도착하는 복수개의 패킷을 수신하고 통신 링크로 패킷을 전송하는 고속 통신 망에서의 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법에 있어서,

각 연결별로 서비스 속도에 비례하는 가중치(W)를 설정하고 상기 가중치를 가용 크레딧(AC)으로 설정하는 제1 단계;

입력되는 적어도 하나의 연결별 패킷을 수신하여 저장하는 제2 단계;

상기 저장된 각 연결의 선두(HOL) 패킷의 나머지 크기(RSP)가 0이면 상기 수신된 패킷의 크기(SP)를 상기 가용 크레딧(AC)의 크기와 비교한 결과에 따라 상기 연결의 패킷에 대한 연결 식별자(ID), 상기 연결의 라운드 수(RN) 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰 큐에 저장하는 제3 단계; 및

상기 토큰 큐에 저장된 토큰이 지정하는 상기 저장된 패킷을 서비스하는 제4 단계를 포함하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.



**【청구항 10】**

제 9항에 있어서,

상기 토큰 큐는 선입선출(FIFO)로 서비스되는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

**【청구항 11】**

제 9항에 있어서, 상기 제3 단계는,

상기 가용 크레딧이 상기 입력 패킷 크기(SP)보다 크거나 같은지를 판단하는 제5 단계;

상기 제5 단계의 판단결과 상기 가용 크레딧(AC)이 상기 입력 패킷 크기(SP)보다 크거나 같으면 해당 연결의 서비스 받을 크레딧 값(CV)을 상기 패킷의 크기(SP)로 설정하고 라운드 수(RN)를 0으로 설정하여 상기 해당 연결의 토큰을 설정하며 상기 가용 크레딧(AC)=(상기 AC - 상기 SP)로 재설정하는 제6 단계;

상기 제5 단계의 판단결과 상기 가용 크레딧(AC)이 상기 입력 패킷 크기(SP)보다 작으면 상기 패킷 크기(SP)와 상기 가용 크레딧(AC)을 감산한 값(SP-AC)과 상기 가중치(W)의 크기를 비교하는 제7 단계;

상기 제7 단계의 비교결과 상기 (SP-AC)의 값이 상기 가중치(W)보다 작거나 같으면 해당 연결의 서비스 받을 크레딧 값(CV)을 상기 가용 크레딧(AC)으로 설정하고 라운드 수(RN)를 1로 설정하여 상기 해당 연결의 토큰을 설정하며 상기 (SP-AC)값을 패킷의 나머지 크기(RSP)로 설정하는 제8 단계;

상기 제7 단계의 비교결과 상기 (SP-AC)의 값이 상기 가중치(W)보다 크면 해당 연결의 서비스 받을 크레딧 값(CV)을 상기 가중치(W)로 설정하고 라운드 수 (RN)=(SP-AC-1)/W 로 설정하여 상기 해당 연결의 토큰을 설정하며 패킷의 나머지 크기(RSP)를 (SP-AC-(RN-1)W)로 재설정하는 제9 단계를 포함하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.;

#### 【청구항 12】

제 11항에 있어서,

각 연결별로 연결 식별자(ID), 가중치(W), 가용 크레딧(AC), 선두(HOL) 패킷 크기(SP), 확보된 크레딧(CC), 백로그된 패킷 크기(BS), 선두(HOL)의 나머지 패킷 크기(RSP)의 파라미터를 관리하여 입력 패킷의 크기(SP)에 따라 재설정되는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

#### 【청구항 13】

제 11항에 있어서,

상기 패킷의 수신 및 패킷의 출력시 상기 패킷의 크기(SP)가 확보된 크레딧(CC)보다 작은 경우 상기 선두(HOL) 패킷의 나머지 크기(RSP)가 0보다 크면 가용 크레딧(AC)이 있더라도 즉시 토큰을 설정하지 않고 다음 토큰을 기다려서 가중치(W)보다 작은 크레딧을 갖는 토큰을 합하여 설정하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

#### 【청구항 14】

제 9항에 있어서,

동일 연결에 대하여 적어도 하나의 토큰이 설정된 경우 가용 크레딧(AC) 범위 내에서 도착하는 순서에 따라 패킷을 서비스하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

**【청구항 15】**

제 9항에 있어서,

상기 제4 단계에서 선두(HOL) 토큰이 지정하는 해당 연결의 선두(HOL) 패킷을 서비스하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

**【청구항 16】**

제 9항에 있어서, 상기 제3 단계는,

상기 저장된 입력 패킷 중 선두(HOL) 패킷의 크기가 선두(HOL) 토큰에 설정된 서비스 받을 크레딧 값(CV)과 같거나 작으면 상기 선두 패킷을 서비스하고 그렇지 않으면 상기 크레딧 값(CV)을 확보된 크레딧(CC)에 추가하고 가용 크레딧(AC) 만큼의 크레딧을 다시 상기 토큰 큐에 저장하는 제10 단계; 및

상기 선두 패킷이 서비스된 후 상기 해당 연결의 백로그된 패킷(BS)이 있거나 새로운 패킷이 도착하면 다시 필요한 만큼의 가용 크레딧(AC)을 할당받아 해당 연결의 토큰을 토큰 큐에 저장하는 제11 단계를 더 포함하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

**【청구항 17】**

제 9항에 있어서,

패킷 출력시 선두(HOL) 토큰의 라운드 수(RN)가 1보다 크면 상기 라운드 수(RN)을 RN-1로 설정하여 다시 토큰 큐에 저장하고 상기 RN이 1이면 상기 RN을 0으로 설정하고 상기 CV를 가중치(W)와 백로그 크기(BS)와 비교하여 작은 값으로 설정하여 다시 토큰 큐에 저장하며 상기 RN이 0이면 확보된 크레딧 크기(CC)와 선두(HOL) 패킷 크기를 비교하여 서비스하는 것을 특징으로 하는 크레딧 기반 라운드 로빈을 이용한 패킷 스케줄링방법.

#### 【청구항 18】

대용량 프로세서를 구비하며 각각의 서비스 속도를 갖는 복수개의 연결로부터 네트워크 스위치에 도착하는 복수개의 패킷을 수신하고 통신 링크로 패킷을 전송하는 고속 통신 망 시스템에,

각 연결별로 서비스 속도에 비례하는 가중치(W)를 설정하고 상기 가중치를 가용 크레딧(AC)으로 설정하는 제1 기능;

입력되는 적어도 하나의 패킷을 수신하여 저장하는 제2 기능;

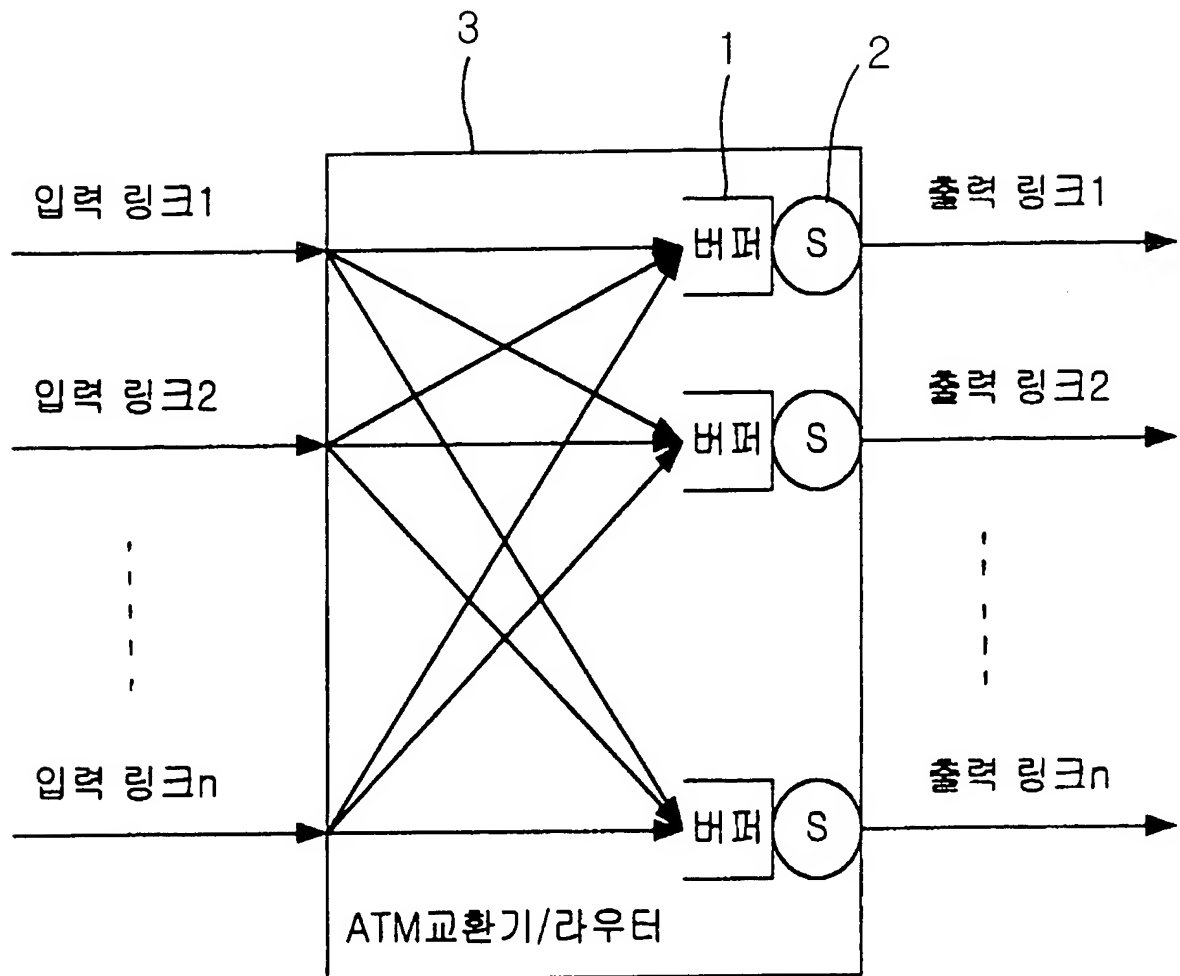
상기 저장된 각 연결의 패킷의 크기(SP)를 상기 가용 크레딧(AC)의 크기를 비교한 결과에 따라 상기 연결의 패킷에 대한 연결 식별자(ID), 상기 연결의 라운드 수(RN) 및 서비스 받을 크레딧 값(CV)을 갖는 토큰을 발생시켜 상기 토큰 큐에 저장하는 제3 기능; 및

상기 토큰 큐에 저장된 토큰이 지정하는 상기 저장된 패킷을 서비스하는 제4 기능;

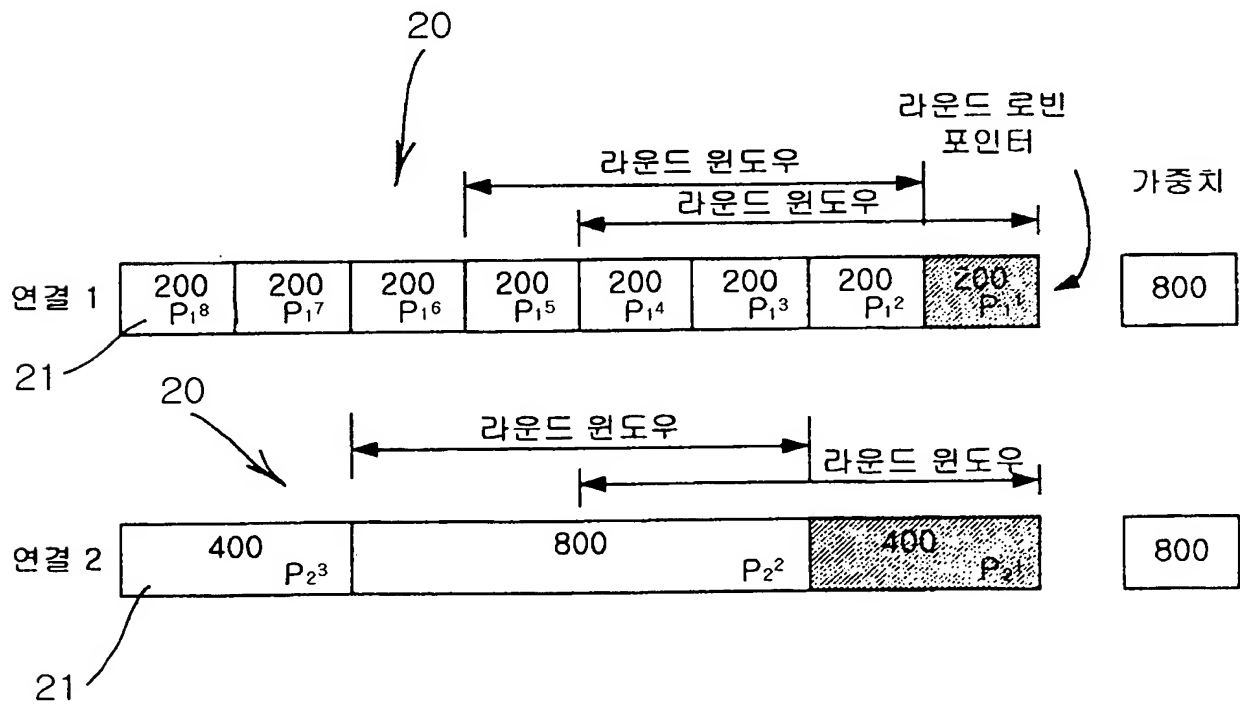
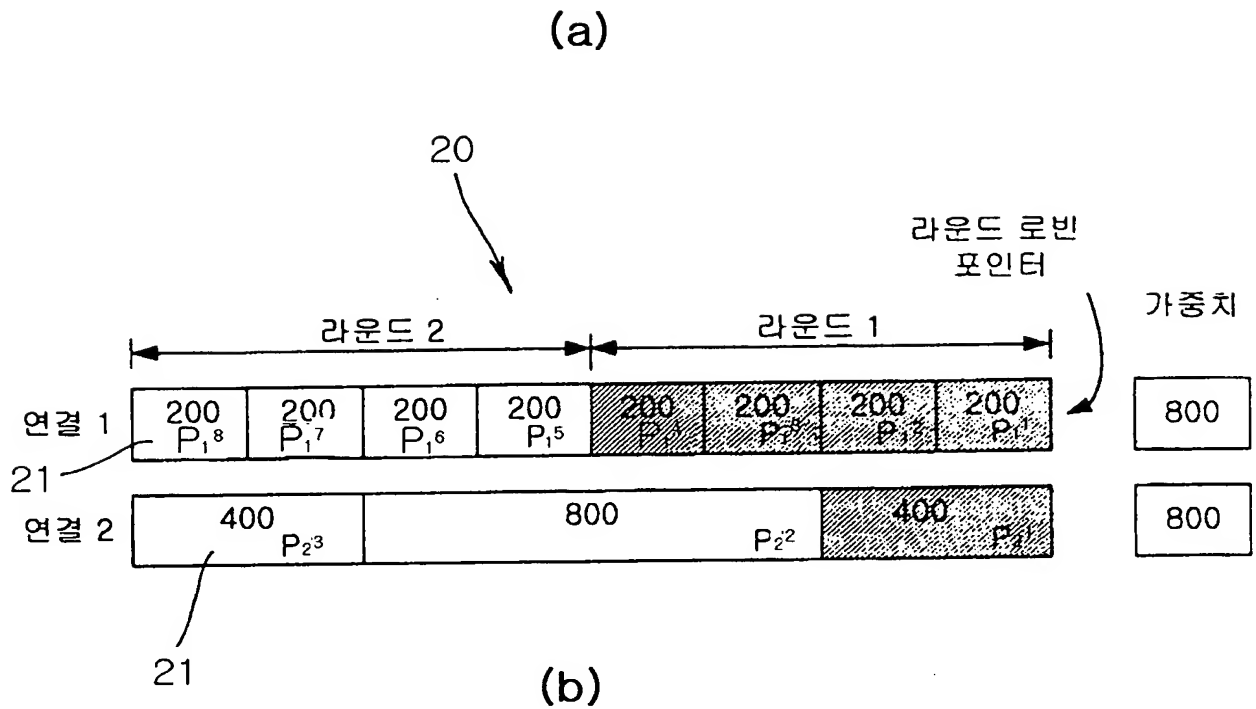
을 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체.

【도면】

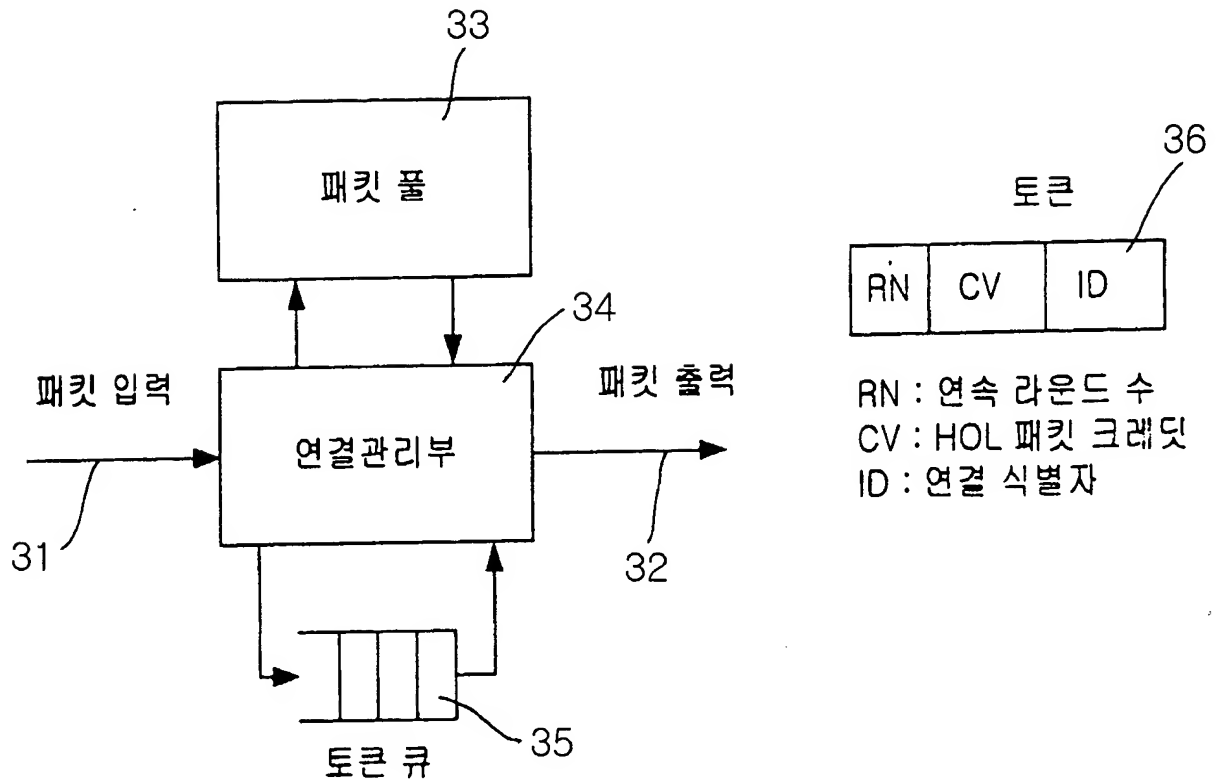
【도 1】



【도 2】



【도 3】



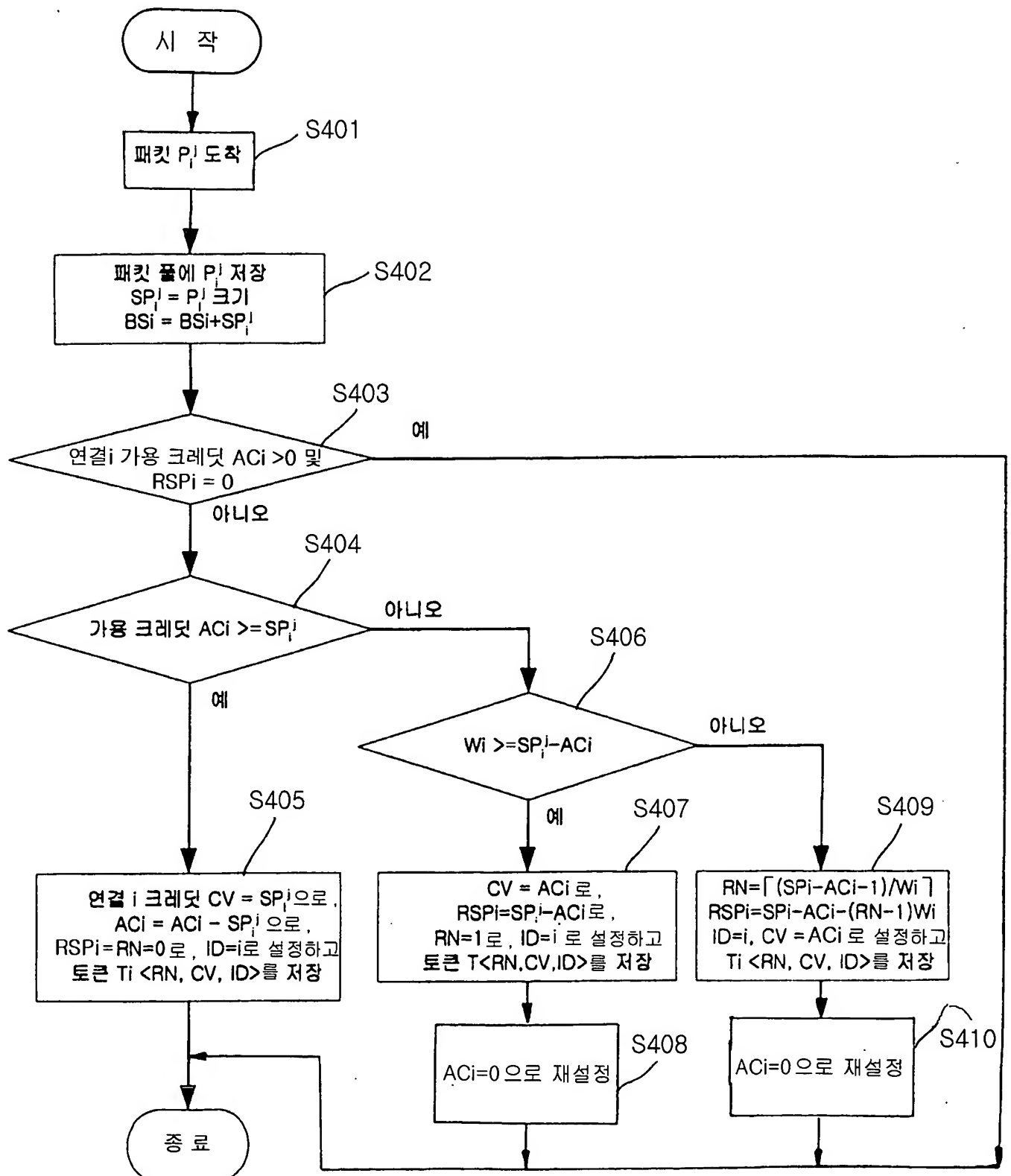
연결 관리 테이블 37

ID	WC	AC	SP	CC	BS	RSP	...
0	100	100	0	0	0	0	...
i	400	200	200	0	200	0	...
...	...	...	...	...	...	...	...
n	200	100	500	200	500	0	...


Legend for connection management table fields:

- WC : 가중 크레딧 (Weighted Credit)
- AC : 가용 크레딧 (Available Credit)
- SP : HOL 패킷 크기 (HOL Packet Size)
- CC : 확보된 크레딧 (Secured Credit)
- BS : 백로그 크기 (Backlog Size)
- RSP : HOL 패킷의 나머지 크기 (Remaining size of HOL packet)

【도 4】



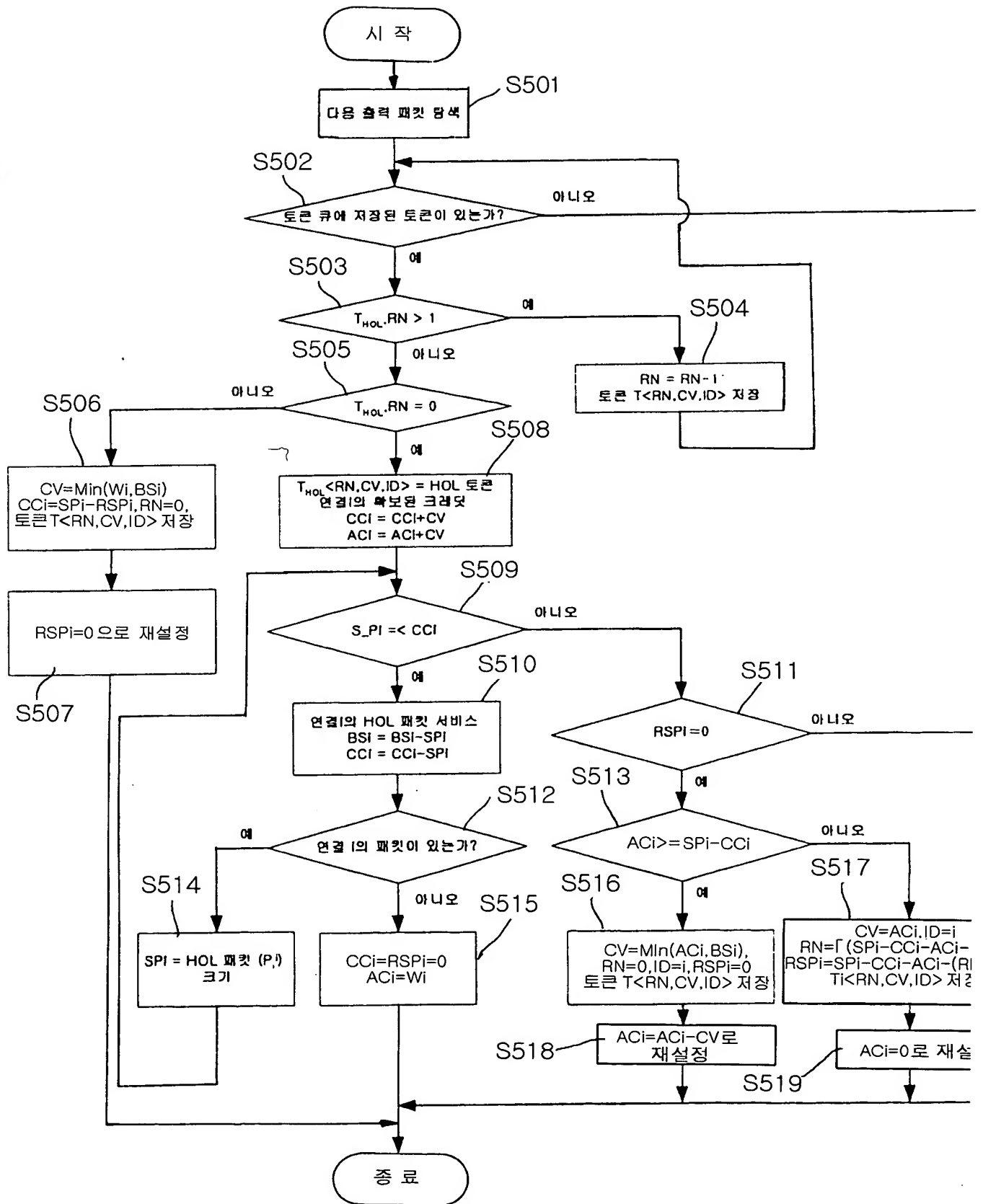





1020010075929

출력 일자: 2002/1/2

【도 5】



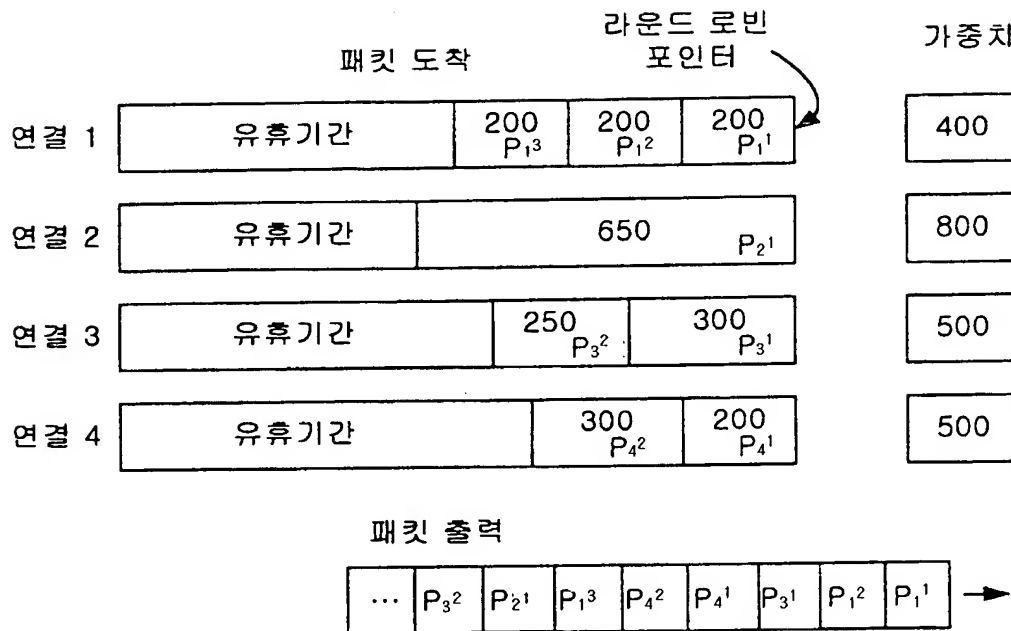


1020010075929

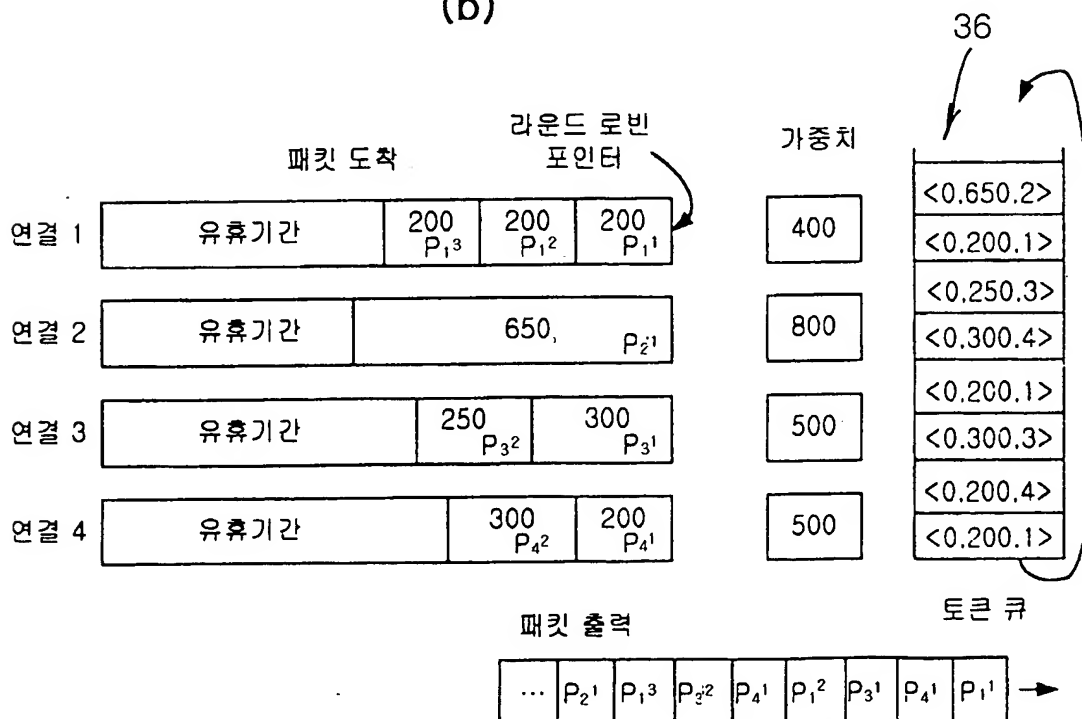
출력 일자: 2002/1/2

【도 6】

(a)



(b)



【도 7】

